


Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S71	172	(715/517-521.ccls.) and interface\$2 same component\$2 <i>Scan all</i>	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/08 15:02
S70	256	(717/120-122.ccls.) and interface\$2 same component\$2	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/08 15:01
S67	119	(select\$4 or associat\$4) adj interfac\$3 near1 component\$2	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/07 17:24
S66	210	(select\$4 or associat\$4) adj interfac\$3 near2 component\$2	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/07 17:24
S63	69	(permission or permitted or authorize or authorized) near4 (insert\$4 or embed\$4) near2 (code or module or interface) <i>Rev all</i>	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/03/16 11:05
S62	25	(permission or permitted) near4 (insert\$4 or embed\$4) near2 (code or module or interface)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/03/16 11:05
S60	105	(permit or permission) same (insert\$4 or embed\$4) near2 interface <i>Scan all</i>	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/03/16 10:47
S56	196	program near2 development and (user\$2 near2 group)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/03/16 08:52
S1	31	program near2 development and plurality near2 interfaces	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/03/16 08:51

S55	105	program near2 development and ((group or plurality) near2 user\$2) same interface\$2 <i>Scan all</i>	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/03/15 17:55
S53	26	program near2 development and plurality near2 interfaces and (group or (users near2 plurality)) <i>Riv all</i>	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/03/15 17:53
S52	26	program near2 development and plurality near2 interfaces and (group or users near2 plurality)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/03/15 17:47




USPTO

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

Search: ☒ The ACM Digital Library ☐ The Guide

component selection interface

THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used component selection interface

Found 66,546 of 160,906

Sort
results
by

Display
results

relevance

expanded form

 [Save results to a Binder](#) [Search Tips](#)☐ Open results in a new windowTry an [Advanced Search](#)
Try this search in [The ACM Guide](#)

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale **1 [Using hypertext in selecting reusable software components](#)**

Michael L. Creech, Dennis F. Freeze, Martin L. Griss

September 1991 **Proceedings of the third annual ACM conference on Hypertext**Full text available:  pdf(1.05 MB)Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)**2 [The COMQUAD component model: enabling dynamic selection of implementations by weaving non-functional aspects](#)**

Steffen Göbel, Christoph Pohl, Simone Röttger, Steffen Zschaler

March 2004 **Proceedings of the 3rd international conference on Aspect-oriented software development**Full text available:  pdf(1.21 MB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The reliability of non-functional contracts is crucial for many software applications. This added to the increasing attention this issue lately received in software engineering. Another development in software engineering is toward component-based systems. The interaction of both, non-functional aspects and components, is a relatively new research area, which the COMQUAD project is focusing on. Our component model, presented in this paper, enables the specification and runtime support of non-func ...

Keywords: AOSD, QoS, adaptivity, components, non-functional properties**3 [Document reuse and semantics: A dynamic user interface for document assembly](#)**

Miro Lehtonen, Renaud Petit, Oskari Heinonen, Greger Lindén

November 2002 **Proceedings of the 2002 ACM symposium on Document engineering**Full text available:  pdf(271.56 KB)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Document assembly has turned out to be a convenient approach to corporate publishing and reuse of large collections of documents. Automated assembly of a document reduces the amount of human effort when creating customized documents consisting of document fragments from a collection. However, most methods used require a number of parameters to be defined prior to the assembly process, and providing these parameters in the correct format is seen to be too demanding for an average user. We have des ...

Keywords: BML, XML, XSLT, document assembly, dynamic user interfaces, structured documents**4 [Automating the lexical and syntactic design of graphical user interfaces: the UofA* UIMS](#)**

Gurminder Singh, Mark Green

July 1991 **ACM Transactions on Graphics (TOG)**, Volume 10 Issue 3Full text available:  pdf(3.82 MB)Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#), [review](#)**5 [Specification of interface components for synchronous data paths](#)**

P. Gutberlet, W. Rosenstiel

May 1994 **Proceedings of the 7th international symposium on High-level synthesis**

Full text available:  pdf(533.14 KB)Additional Information: [full citation](#), [references](#), [citations](#)**6 WREN—an environment for component-based development**

Chris Lüer, David S. Rosenblum

September 2001

ACM SIGSOFT Software Engineering Notes , Proceedings of the 8th European software engineering conference held jointly with 9th ACM SIGSOFT international symposium on Foundations of software engineering, Volume 26 Issue 5Full text available:  pdf(590.37 KB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Prior research in software environments focused on three important problems---tool integration, artifact management, and process guidance. The context for that research, and hence the orientation of the resulting environments, was a traditional model of development in which an application is developed completely from scratch by a single organization. A notable characteristic of component-based development is its emphasis on integrating independently developed components produced by multiple orga ...

Keywords: Java, Java Beans, component-based software engineering, software components, software environments

7 Multiple mass-market applications as components

David Coppit, Kevin J. Sullivan

June 2000

Proceedings of the 22nd international conference on Software engineeringFull text available:  pdf(179.51 KB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Truly successful models for component-based software development continue to prove elusive. One of the few is the use of operating system, database and similar programs in many systems. We address three related problems in this paper. First, we lack needed models. Second, we do not know the conditions under which such models can succeed. In particular, it is unclear whether the notable success with operating systems can be replicated. Third, we do not know whether certain specific models ca ...

Keywords: component-based software, package-oriented programming

8 Poster session 2: ICARE software components for rapidly developing multimodal interfaces

Jullien Bouchet, Laurence Nigay, Thierry Ganille

October 2004

Proceedings of the 6th international conference on Multimodal interfacesFull text available:  pdf(665.26 KB)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)


Although several real multimodal systems have been built, their development still remains a difficult task. In this paper we address this problem of development of multimodal interfaces by describing a component-based approach, called ICARE, for rapidly developing multimodal interfaces. ICARE stands for Interaction-CARE (Complementarity Assignment Redundancy Equivalence). Our component-based approach relies on two types of software components. Firstly ICARE elementary components include Devic ...

Keywords: multimodal interactive systems, software components

9 Communication technologies: A new generation of communication aids under the ULYSSES component-based framework

Georgios Kouroupetroglou, Alexandros Pino

July 2002

Proceedings of the fifth international ACM conference on Assistive technologiesFull text available:  pdf(1.21 MB)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)


In this paper, we introduce a new generation of computer-based communication aids, designed and developed using state of the art software engineering models and architectures. The communicators we present are based on a component-based framework called ULYSSES that aims to simplify the integration of multi-vendor components into low cost products and maximizes modularity and reusability. Following the ULYSSES approach, one can build up powerful and reliable applications, adaptable to various use ...

Keywords: Augmentative and Alternative Communication (AAC), communication aids, communicators, component based development, framework architecture

10 Virtual body language: providing appropriate user interfaces in collaborative virtual environments

Jolanda Tromp, Dave Snowden

September 1997

Proceedings of the ACM symposium on Virtual reality software and technologyFull text available:  pdf(1.11 MB)Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)**11 Selectors: going beyond user-interface widgets**

Jeff Johnson

June 1992

Proceedings of the SIGCHI conference on Human factors in computing systemsFull text available:  pdf(1.03 MB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Most UI toolkits and UIMSs make use of widgets, e.g., buttons, text fields, sliders, menus. Designers construct user interfaces by choosing and laying out widgets, then connecting them to application semantics. This approach has four problems. First, most widgets are too low-level; constructing interfaces from them takes too much work. Second, working with widgets focuses attention on appearance and layout issues, rather than on more important semantic design issues. Third, designers can ea ...

Keywords: UIMS, user-interface toolkit, widgets

12 A unidraw-based user interface builder

John M. Vlissides, Steven Tang

October 1991

Proceedings of the 4th annual ACM symposium on User interface software and technologyFull text available:  pdf(1.40 MB)Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)**13 An object-oriented approach to graphical interfaces**

Paul S. Barth

April 1986

ACM Transactions on Graphics (TOG), Volume 5 Issue 2Full text available:  pdf(2.23 MB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

An object-oriented system for building graphical interfaces to programs is discussed. The system, called GROW, facilitates the process of creating interfaces that are highly interactive (including direct manipulation and animation), rich in layout structure, and effectively reusable across applications. These properties are achieved through three techniques: object-based graphics with taxonomic inheritance, interobject relationships such as composition and graphical dependency, and separati ...

14 Component specialization

Gustavo Bobeff, Jacques Noyé

August 2004

Proceedings of the 2004 ACM SIGPLAN symposium on Partial evaluation and semantics-based program manipulationFull text available:  pdf(285.59 KB)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Component-Based Software Development (CBSD) is an attractive way to deliver generic executable pieces of program, ready to be reused in many different contexts. Component reuse is based on a black-box model that frees component consumers from diving into implementation details. Adapting a generic component to a particular context of use is then based on a parameterized interface that becomes a specific component wrapper at runtime. This shallow adaptation, which keeps the component implementation ...

Keywords: component generator, component-based software development, partial evaluation, program slicing

15 Modeling methodology: Architectures and languages for model building and reuse: organization and selection of reconfigurable models

Antonio Diaz-Calderon, Christiaan J. J. Paredis, Pradeep K. Khosla

December 2000

Proceedings of the 32nd conference on Winter simulationFull text available:  pdf(314.60 KB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

This paper introduces the concept of reconfigurable simulation models and describes how these models can be used to support simulation-based design. As in object-oriented programming, a reconfigurable model consists of a separate interface and multiple implementations. An AND-OR tree represents which implementations can be bound to each interface. From the resulting model space, a designer can quickly select the simulation model that is most appropriate for the current design stage. We conclude ...

16 Contigra: an XML-based architecture for component-oriented 3D applications

Raimund Dachsel, Michael Hinz, Klaus Meißner

February 2002 **Proceeding of the seventh international conference on 3D Web technology**

Full text available:  [pdf\(368.21 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


Even though numerous Web3D technologies exist, most of them do not support a high-level, multi-disciplinary authoring process. Moreover, concepts of reuse are rarely provided. A component-based approach is introduced with the CONTIGRA architecture to construct interactive, three-dimensional applications, either stand-alone or web-based. The approach is entirely based on declarative XML documents describing the component implementation, its interface, as well as component configuration and compo ...

Keywords: 3D components, 3D user interfaces, 3D widgets, XML schema, component-based development, contigra, extensible 3D (X3D), virtual environments

17 XXL: a dual approach for building user interfaces

Eric Lecolinet

November 1996 **Proceedings of the 9th annual ACM symposium on User interface software and technology**

Full text available:  [pdf\(1.96 MB\)](#)


Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

Keywords: distributed interfaces, interface builders, iterative development, scripting languages, textual and visual equivalence, user interface software

18 Interfaces to databases: A framework for user-interfaces to databases

Kenneth J. Mitchell, Jessie B. Kennedy, Peter J. Barclay

May 1996 **Proceedings of the workshop on Advanced visual interfaces**

Full text available:  [pdf\(1.56 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#)


A framework for user-interfaces to databases (IDSs) is proposed which draws from existing research on human computer interaction (HCI) and database systems. The framework is described in terms of a classification of the characteristic components of an IDS. These components, when progressively refined, may be mapped to a conceptual object-oriented language for the precise specification of the IDS. A prototype system is presented, showing the potential for automated mapping of a language specifica ...

Keywords: conceptual modelling, direct manipulation interfaces, human-computer interaction (HCI), user-interfaces to databases

19 Chisel: a system for creating highly interactive screen layouts

G. Singh, M. Green

November 1989 **Proceedings of the 2nd annual ACM SIGGRAPH symposium on User interface software and technology**

Full text available:  [pdf\(1.12 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The UofA* User Interface Management System (UIMS) generates graphical user interfaces based on a high-level description of semantic commands supported by the application. A main part of the UIMS, called Chisel, generates the presentation component of interfaces. Chisel selects interaction techniques, determines their attributes, and places them on the screen of the display device. While doing so it is capable of considering device properties, end user's preferences, and ...

20 Component selection and matching for IP-based design

G. Martin, R. Seepold, T. Zhang, L. Benini, G. De Micheli

March 2001 **Proceedings of the conference on Design, automation and test in Europe**

Full text available:  [pdf\(170.22 KB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

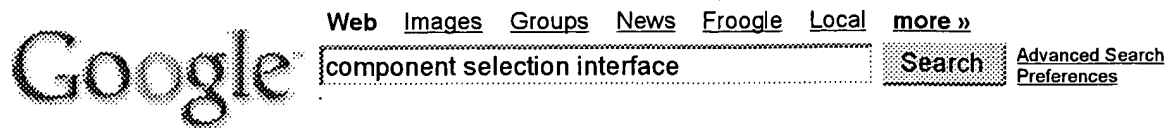
Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)

**Web**Results 1 - 10 of about 10,000,000 for **component selection interface**. (0.24 seconds)**IDesignerEventService Interface (.NET Framework)**

A root designer provides design time support for the base **component** of a ... event provides notification when the current **component selection** has changed. ...
 msdn.microsoft.com/.../en-us/cpref/html/ flrlfsystem**component**modeldesignidesignereventserviceclasstopic.asp - 28k - Sep 6, 2005 - [Cached](#) - [Similar pages](#)

Chapter 4 Installing Software Using the Graphical Interface

Proceed through the installer pages to the **Component Selection** page. Click View **Component** Statuses at the ... Start the graphical installation **interface**: ...
 docs.sun.com/source/817-5760/ins-gui.html - 62k - [Cached](#) - [Similar pages](#)

Chapter 5 Installing Software Using the Graphical Interface

Proceed through the installer pages to the **Component Selection** page. ... Start the graphical installation **interface**: ./installer ...
 docs.sun.com/source/816-6874/std-ins-gui.html - 65k - [Cached](#) - [Similar pages](#)

Pentium® 4 Processor-based Motherboards - Integration Guide ...

Information on choosing **components** when building your own PC. ... It is critical to **select** a chassis that supports the chosen board size. ...
 support.intel.com/support/ motherboards/desktop/sb/cs-020835.htm - 38k - [Cached](#) - [Similar pages](#)

Desktop Boards - Integration Guide: Component Selection

Integration Guide: **Component Selection** ... Intel Desktop Boards include the following: a diskette drive **interface**, two PS/2 ports, one serial port, ...
 support.intel.com/support/motherboards/desktop/ sb/CS-012555.htm?iid=asmo-na_vpe_prodSelect_sysComponents& - 40k - [Cached](#) - [Similar pages](#)

Component Interface

Using the **component interface**, your program can call PC-SPAN, specify the updated ... Sets currently **selected** point in time to NULL (removes **selection**). ...
 www.cme.com/clearing/rmspan/ sog/onlinegd/compint3073.html - 54k - [Cached](#) - [Similar pages](#)

Frequency component selection for an EEG-based brain to computer ...

Frequency **component selection** for an EEG-based brain to computer **interface**. ... In this study the relevance of different spectral **components** is analyzed: 1) ...
 www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve& db=PubMed&list_uids=10609628&dopt=Abstract - [Similar pages](#)

Frequency component selection for an ECoG-based brain-computer ...

Frequency **component selection** for an ECoG-based brain-computer **interface**.
 Scherer R, Graimann B, Huggins JE, Levine SP, Pfurtscheller G. ...
 www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve& db=PubMed&list_uids=12655847&dopt=Abstract - [Similar pages](#)

GraphMaker - Requirements Specification

The menu **interface** shall consist of a menu bar at the top of the program window. ... **Component selection** standards. Specify a vertex by using a mouse. ...
 www.blumars.com/java/graphmaker/specs/req_spec.html - 22k - Sep 7, 2005 - [Cached](#) - [Similar pages](#)

[PDF] Frequency component selection for an EEG-based brain to computer ...

File Format: PDF/Adobe Acrobat
interface (BCI). Such a system classifies electrical brain signals. online. ...
components. The new results suggest that frequency **selection** ...
 ieeexplore.ieee.org/iel5/ 86/17527/00808944.pdf?arnumber=808944 - [Similar pages](#)

Result Page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [Next](#)



Free! Instantly find your email, files, media and web history. [Download now.](#)

Search

[Search within results](#) | [Language Tools](#) | [Search Tips](#) | [Dissatisfied? Help us improve](#)

[Google Home](#) - [Advertising Programs](#) - [Business Solutions](#) - [About Google](#)

©2005 Google